특집 01

사이버 복원력을 고려한 유무인 복합체계의 참조 아키텍처 설계

김동환 (LIG넥스원)

목 차

1. 서 론
2. 무기체계 개발방법론 및 문제점
3. 참조 유무인 복합체계의 요구사항
4. 유무인 복합체계의 참조 아키텍처 설계
5. 결 론

1. 서 론

현대전에서의 유무인 복합체계는 무기체계의 여러 형태중의 하나가 아니라 궁극의 목표가 되는 필수적인 메가 트렌드(Mega Trend)이다. 유무인 복합체계는 기술적으로는 인공지능 소프트웨어를 적용하여 MUM-T(Manned-Unmanned Teaming)의 개념을 구현한 체계이며[1], 전술적으로는 작전시간 동안 동적으로 킬체인을 변경하는 모자이크전(Mosaic Warfare)을 실현하는 물리적 실체라고 할 수 있다[2].

인공지능 기반의 소프트웨어 중심 체계인 유무인 복합체계의 이러한 특징은 반대로 사이버 복원력(Cyber Resilience)을 고려한 개발이 필수적으로 요구된다. 이러한 이유 때문에 무기체계 개발 시 RMF(Risk Management Framework)[3]를 적용하기 위한 논의들이 이루어지고 있지만, 전통적으로 무기체계 개발은 높은 신뢰성을 요구하는 System-of-Systems을 개발하는 엄격한 절차와 기준을 포함하고 있다.

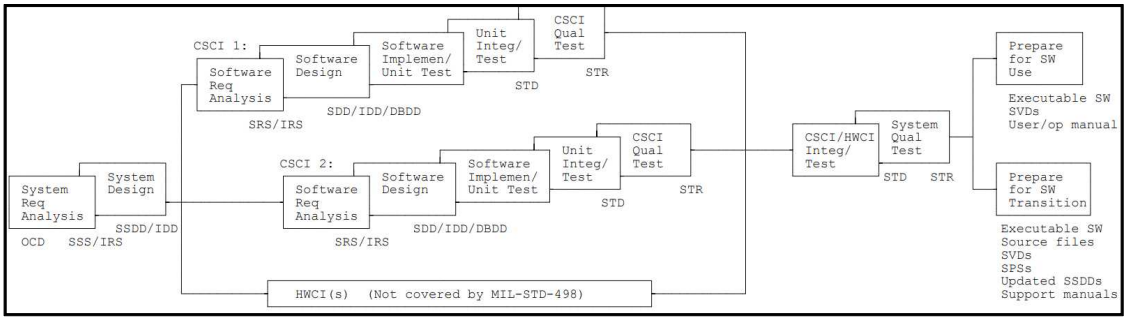
본 논문에서는 MBSE(Model-Based System Engineering) 기반의 무기체계 개발방법론을 적용하여 사이버 복원력을 고려한 유무인 복합체계의 참조 아키텍처를 제안하고자 한다. 이를 통해 향후 개별적인 유무인 복합체계를 개발할 때 공통적으로 적용할 수 있는 가이드라인을 제시하고자 한다.

2장에서는 참조 아키텍처 개발을 위한 무기체계 개발방법론 및 문제점을 소개한다. 3장에서는 사이버 복원력을 고려한 참조 유무인 복합체계의 요구사항을 기능 및 비기능적 요구사항으로 구분하여 제안한다. 4장에서는 사이버 복원력을 포함한 다양한 품질속성을 반영한 최종 참조 아키텍처를 제안한다. 마지막으로 5장에서는 본 논문의 결론과 향후 연구 방향에 대해 제시하고자 한다.

2. 무기체계 개발방법론 및 문제점

2.1 무기체계 개발 프로세스의 소개 및 문제점

국내 무기체계 개발 프로세스는 MIL-STD-498[4]을 기반으로 하고 있다. MIL-STD-498은 현재 ISO 12207[5]에 통합되었지만, De-facto표준으로



(그림 1) MIL-STD-498의 무기체계 개발 프로세스

서 유럽과 미국의 방산업체에서 무기체계 개발에 적용하고 있다. (그림 1)은 MIL-STD-498에 기술된 대표적인 개발프로세스를 표현하고 있다. MIL-STD-498은 소프트웨어 시스템을 기반으로 개발 프로세스를 정의하고 있어 소프트웨어와 하드웨어가 통합된 체계관점에서의 분석으로부터 시작된다.

개발 프로세스는 체계분석 및 설계를 통해 체계를 구성하는 소프트웨어 형상항목(CSCI: Computer Software Configuration Item)과 하드웨어 형상항목(Hardware Configuration Item)을 식별하고 이들 간의 정적 및 동적인 관계를 설계하여 최종적으로 체계 아키텍처라는 산출물을 개발하도록 권고하고 있다. 그리고 각 형상항목별로 분석설계를 통해 최종 구현물을 제작하고 통합하여 시험평가를 실시하는 프로세스를 진행하며 개발이 진행된다.

MIL-STD-498에서는 이러한 절차들과 주요한 가이드라인을 포함하며, 가장 일반적인 폭포수형(Water-fall) 라이프사이클모델에서부터 점진적(Incremental), 진화적(Evolutionary) 라이프사이클모델까지 다양한 라이프사이클모델을 적용할 수 있다. 그러나 국내 무기체계 개발 매뉴얼[6]은 MI-STD-498을 기반으로 개발되었음에도 사이버 복원력과 같은 품질속성을 반영하기에는 여러가지 한계가 존재한다[7].

2.1.1 빈약한 요구사항의 관리

요구사항을 결정하는 것은 개발에서 있어서 가장 어려운 이슈 중에 하나이다. 시스템개발 프로젝트가 실패하는 원인으로 요구사항 관리의 실패가 큰 것으로 알려져 있다. 요구사항은 기능적 요구사항(Functional Requirements)과 품질속성으로 표현되는 비기능적 요구사항(Non-Functional Requirements)으로 크게 분류할 수 있다. 비기능적 요구사항은 기능을 수행하는 데 있어서 어느 수준으로 개발할 것인지에 대한 요구사항으로 품질에 가장 직접적인 영향을 주는 요구사항이다.

그러나 국내 무기체계 매뉴얼에서는 기능적 요구사항의 완전성과 일관성을 보장하기 어려운 문서 및 특정 사례 중심의 가이드라인을 제공한다. 또한 기능적 요구사항 분석에만 치중하고 비기능적 요구사항에 대한 분석 및 정의에는 매우 취약하다. 특히 명시적이지 않은 비기능 요구사항은 거의 정의되지 않아 테스트단계에서 이러한 정의되지 않은 요구사항이 새롭게 식별되어 최종적으로 구현된 시스템에 보이지 않는 결함을 내포하는 경우가 존재한다.

2.1.2 형식적인 품질관리

무기체계는 매우 높은 수준의 품질이 요구된다. 따라서 무기체계의 품질을 명확히 정의하고 엄격

하게 품질을 검증하고 확인하는 것은 매우 중요하다. 특히 요구사항을 검증 및 확인(Verification and Validation)이 가능하도록 정의하는 것은 품질관리에서 매우 중요하다. 따라서 IV&V(Independent V&V)를 통한 품질관리 활동을 강조하고 있다.

그러나 국내 무기체계 매뉴얼은 문서 존재유무 및 정성적인 체크리스트를 통해 주관적인 품질관리가 이루지고 있어 품질의 정확한 관리가 매우 미흡하다. 실제 구현된 결과가 나오기 전까지는 개발과정에서의 시스템의 품질을 평가하는 것은 매우 어려운 실정이다. 문서 템플릿을 통해 상세한 가이드라인을 제공하고는 있지만 사실상 제공된 문서 템플릿이 한정된 분야에 국한되어 있고 무기체계에 적용하기 어려워 템플릿에 맞추다 보면 오히려 본질에서 벗어나 품질이 오히려 저하되는 경우가 많다.

2.1.3 기술부채의 증가

러시아-우크라이나 전쟁을 통해 상용기술의 신속한 활용이 승리의 요인이 되는 것을 직접 경험하였다. 전쟁에서 상용 드론의 활용, 저궤도 상용위성을 이용한 정보 수집 및 통신, 모바일 앱을 통한 정보 교환 등은 기존의 전통적인 전쟁의 개념을 바꾸어 놓았다. 이것은 기술을 신속하게 전쟁에 활용해야 한다는 교훈을 우리에게 주고 있다. 특히 사이버공격에 대응하는 무기체계의 사이버 복원력이 OTA(Over-The-Air)를 통해 신속히 유지되는 것을 보면서 기술이 전장에 신속히 적용되는 것이 중요한 성공요인임을 알 수 있었다.

그러나 우리의 무기체계 개발은 전통적인 폭포수형 라이프사이클 모델에만 의존하고 있어 장시간의 개발기간이 소요되며, 운영 측면에서도 개발과 동일한 수준의 성능 개량 시간이 소요되어 기술부채가 매우 높은 무기체계를 개발하고 있다고 판단된다. 특히 기술문서 중심의 개발은 신속한 유지보수의 걸림돌로 작용하고 있으며, 동작하는 System Model을 기반으로 한 개발 및 유지보수가 중요하게 인식되고 있다.

2.2 PRISM 개발방법론

PRISM(Productive, Reliable and Intelligent Software Methodology)은 무기체계 개발매뉴얼의 문제점을 보완하기 위해 개발된 LIG넥스원의 소프트웨어 중심 무기체계 개발방법론이다[8]. MIL-STD-498의 무기체계 개발프로세스에 따라 다양한 라이프사이클모델을 지원하는 Model-Based System Engineering을 적용한 방법론으로 정량적으로 품질을 측정할 수 있도록 개발되었다.

PRISM은 SysML[9] 및 UML[10]이라는 모델링언어들을 이용하여 체계분석에서부터 소프트웨어 설계 까지의 모든 단계를 정형화된 모델을 개발하는 모델링 가이드라인을 제시함으로써 MBSE를 지원한다. 본 논문에서는 PRISM을 적용하여 사이버 복원력을 고려한 아키텍처를 제안하며, 참조 아키텍처의 개발의 범위를 체계 아키텍처의 정적인 구조를 설계하기 위한 체계분석 및 설계 모델링으로 제한하여 소개한다.

3. 참조 유무인 복합체계의 요구사항

유무인 복합체계는 전투를 수행하기 위한 전투체계의 한 형태이다. 따라서 일반적인 작전주기를 설명하는 OODA(Observe, Orient, Decide, Act) Loop를 기반으로 우주, 공중, 지상, 해양 및 사이버의 5대 작전영역에서 공통적으로 적용할 수 있는 전형적인 킬체인을 기반으로 유무인 복합체계의 공통 요구사항을 도출하였다[11].

3.1 기능적 요구사항의 모델링

3.1.1 Context Model

체계 요구사항 분석은 시스템의 범위를 결정하는 것으로 시작한다. 시스템의 범위를 모델링한 분석모델이 Context Model이다. (그림 2)는 클래스 다이어그램을 사용하여 개발된 참조 유무인 복합체계의 Context Model이다.

흑백의 클래스로 표현된 “MUMT system”이 참조 유무인 복합체계를 의미하며 주변의 클래스들은 시스템의 액터(actor)로서 유무인 복합체계와 상호작용하는 외부 객체들이다. 그리고 상호작용은 객체 간의 관계(association)로 표현한다. 관계는 외부 객체 들과의 인터페이스를 표현하며, 인터페이스를 통한 입출력되는 정보 또는 신호를 관계명으로 Interface Model에 별도로 모델링한다.

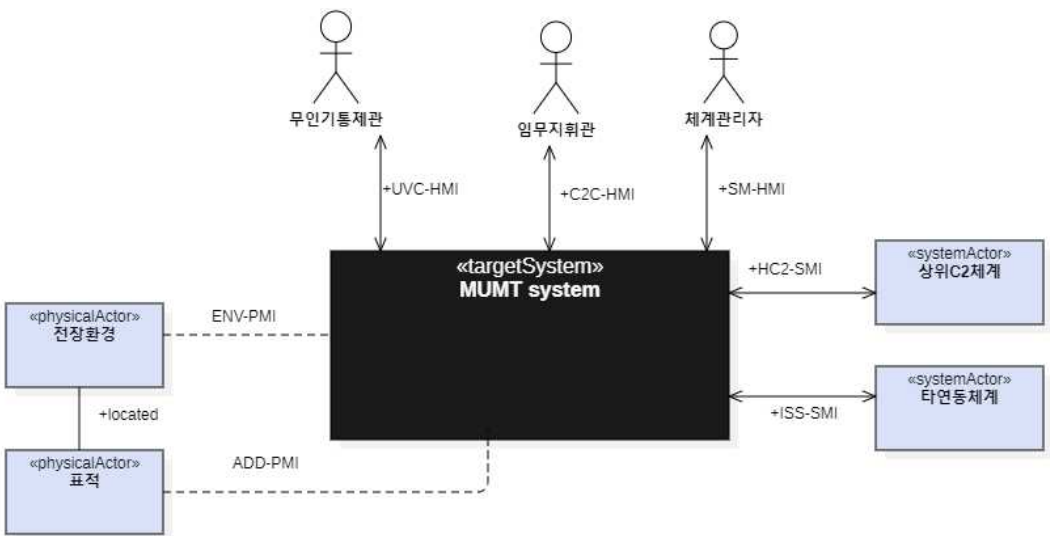
휴먼액터로 유무인 복합체계를 운영하는 운용자로서의 무인기통제관과 통제관에게 임무를 하달하는 상위 지휘관을 식별하였으며 시스템의 기술적 운영을 위한 체계관리자를 식별하였다. 그리고 연동하는 시스템을 2가지로 구분하여 지휘구조에 따른 상위C2(지휘통제)체계와 지휘구조와 무관하게 연동하는 연동체계로 구분하였다. 나머지는 외부 환경 및 적의 위협을 표적이라는 객체로 모델링하였다.

3.1.2 Capability Model

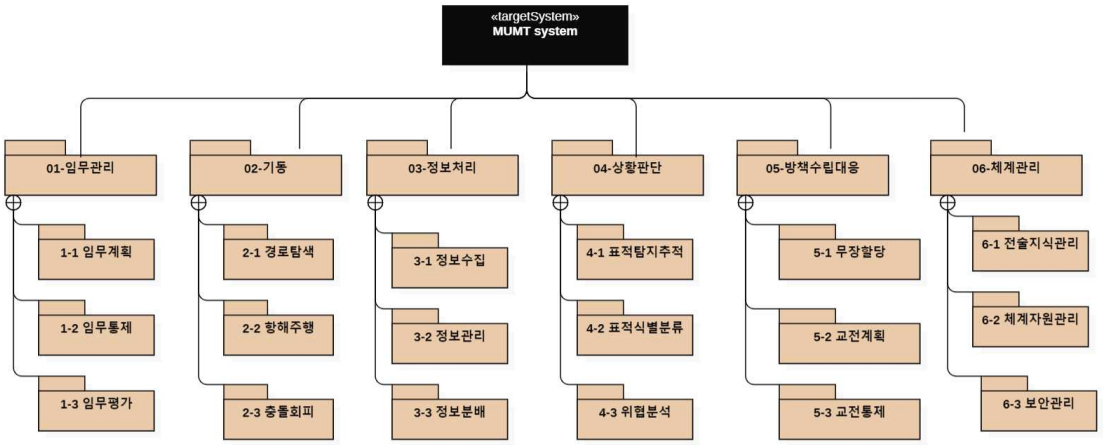
Capability Model은 체계가 수행할 기능 요구사항을 정의한 모델이다. 일반적으로 체계의 규모에 따라 체계수준에서의 최하위 기능을 결정하고 이들을 논리적으로 밀접한 관련이 있는 기능들을 통합하는 추상화를 통해 기능구조를 분석하고, 최하위 기능에 대해서는 세부적인 비즈니스 규칙을 유스케이스 시나리오(Use Case Scenario)로 정의한다. 일반적으로 추상화수준은 3단계로 제한하여 논리적 기능구조를 개발한다.

(그림 3)은 패키지 다이어그램으로 참조 유무인 복합체계의 기능구조를 표현한 Capability Model을 나타내고 있다. 최하위 패키지에는 하나 이상의 단위 기능들을 포함할 수 있다.

최상위의 추상화 기능들 중 정보처리, 상황판단, 방책수립대응은 킬체인을 구성하는 체계의 가



(그림 2) 참조 유무인복합체계의 Context Model



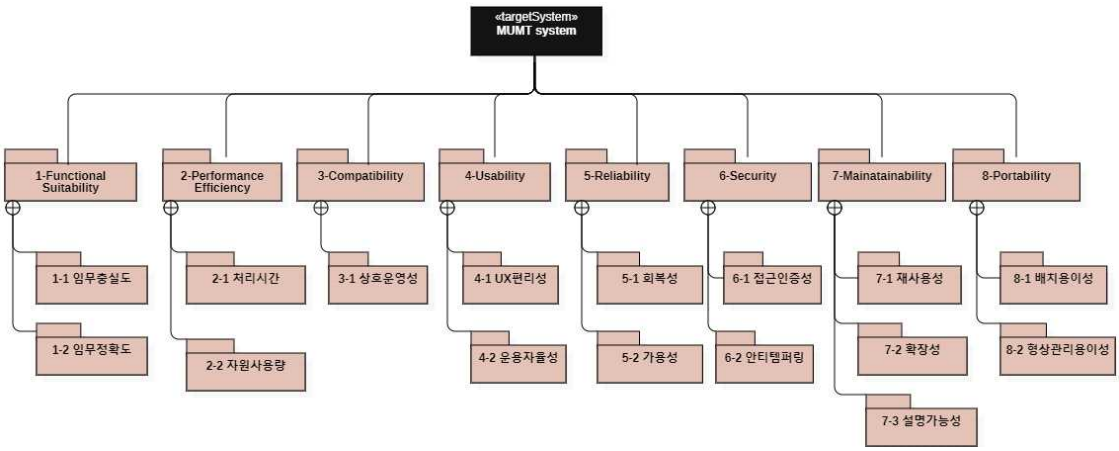
(그림 3) 참조 유무인 복합체계의 논리적 기능구조

장 핵심적인 전투기능이다. 정보처리는 외부로부터의 표적을 관찰하는 Observe의 기능, 상황판단은 표적을 식별하고 위협의 우선순위를 결정하는 Orient의 기능, 방책수립 대응은 교전을 위해 필요한 결정을 내리고 표적과의 교전을 수행하는 Decide와 Act의 기능을 의미한다.

그리고 임무관리는 임무 지휘관의 역할을 수행하는 기능들, 기동은 임무를 수행하는 무인체계들이 이동과 관련한 기능들, 체계관리는 체계관리자를 지원하는 주요 기능들을 식별한 것이다.

3.3 품질 요구사항의 모델링

사이버 복원력과 직접적인 관련이 있는 신뢰성(Reliability) 및 보안성(Security)을 비롯하여 간접적으로 관계를 갖는 품질요구사항을 ISO 25000[12]의 8개의 품질속성을 기준으로 (그림 4)와 같이 패키지 다이어그램으로 모델링하였다. 사이버 복원력을 확보하기 위해서는 단순히 체계가 갖는 고유한 속성 뿐 만 아니라 체계를 운영 및 유지보수하는 관리적인 능력도 중요한 요인으로 인식되고 있다.



(그림 4) 참조 유무인 복합체계의 Quality Model

이러한 관점에서 8개의 제품과 관련된 품질의 대부분이 사이버 복원력과 직간접적으로 관계를 갖고 있다고 판단된다. 예를 들어 유지보수용이성(Maintainability)과 이식성(Portability)은 운용중인 시스템의 신속한 개선과 배포를 지원하여 급변하는 위협으로부터 체계의 생존성을 유지할 수 있는 중요한 속성이라고 할 수 있다. 최하위 기능의 경우 유스케이스시나리오를 작성하는 것과 같이 최종 품질속성에 대해서는 품질속성 시나리오(Quality Attribute Scenario)를 작성한다.

4. 유무인 복합체계의 참조 아키텍처 설계

4.1 사이버 복원력을 고려한 설계결정사항

아키텍처의 설계를 성공적으로 수행하는 것은 검증된 기존의 아키텍처 패턴(Pattern) 또는 스타일(Style)을 적용하는 것이다. 다시 말해 설계란 품질요구사항을 만족하기 위해 어떤 아키텍처 패턴을 적용할 것인가를 결정하거나 설계 가이드라인을 기준으로 새로운 아키텍처 패턴을 식별하는 것이라 할 수 있다.

<표 1>은 목표로 하는 참조 아키텍처를 결정하기 위한 설계결정사항을 정리한 표이다. 설계결정사항을 개발한다는 것은 설계의 모든 의사결정을 수행하는 것이기 때문에 가장 중요한 활동이며 아키텍처는 이러한 설계결정사항의 부산물이라고 해도 과언이 아니다.

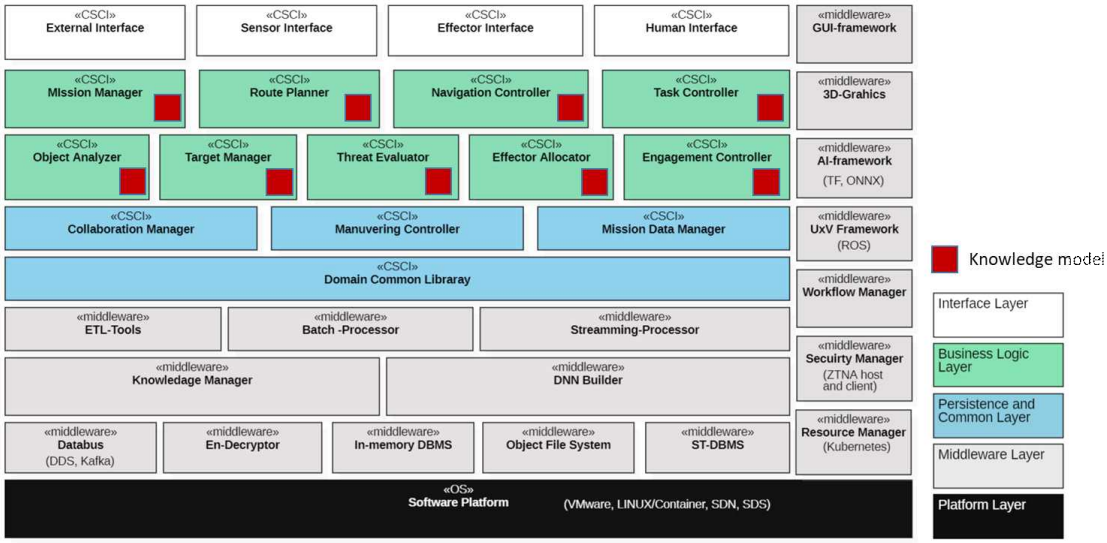
4.2 유무인 복합체계의 참조 아키텍처

4.2.1 소프트웨어 아키텍처

(그림 5)은 설계결정사항에 따라 최종 개발된 소프트웨어 레이어 아키텍처를 표현하고 있다. 제안한 아키텍처에는 본 논문에서 언급되지 않은 설계결정사항의 내용도 포함되어 있으나, 기본적으로는 크게 플랫폼 레이어, 미들웨어 레이어, 공통 및 데이터 담당 레이어, 임무서비스 순수기능을 담하는 비즈니스 로직 레이어, 그리고 액터들과의

〈표 1〉 참조 아키텍처 설계를 위한 설계결정사항

아키텍처	구분	설계항목	설계결정사항	관련 품질속성
기술 아키텍처	컴퓨팅 플랫폼	임무분산구조	임무의 유형을 4개(coordination, collaboration, swarming, team)로 구분하여 분산구조 설계 [13]	운용자율성, 가용성, UX(User Experience)편리성
		SW 플랫폼	개방형플랫폼으로 쿠버네티스와 연동되는 Embedded LINUX Container 기반 플랫폼으로 설계[14]	재사용성, 회복성, 확장성, 배치용이성
		보안 플랫폼	TPM(Trust Platform Module) 또는 HSM(Hardware Security Module)[15]	접근인증성, 안티템퍼링, 상호운영성
	미들웨어 및 프레임워크	데이터버스	실시간 Service-Oriented Architecture 패턴을 지향하는 DDS(Data Distribution Service) 적용[16]	확장성, 회복성 및 상호운영성, 실시간처리
		무인화 프레임워크	로봇분야에서의 개방형 프레임워크인 ROS2 적용[17]	확장성, 재사용성, 상호운영성, 임무정확도
		보안 프레임워크	Zero-trust architecture로 Software Defined Perimeter 구현[18]	접근인증성, 임무충실도, 상호운영성
응용 아키텍처	응용프로그램	레이어 아키텍처	논리적 기능구조를 기반으로 Interface layer, Business layer, Common 및 Persistence Layer를 구성	재사용성, 확장성, 형상관리용이성, 배치용이성



(그림 5) 참조 유무인 복합체계의 소프트웨어 아키텍처

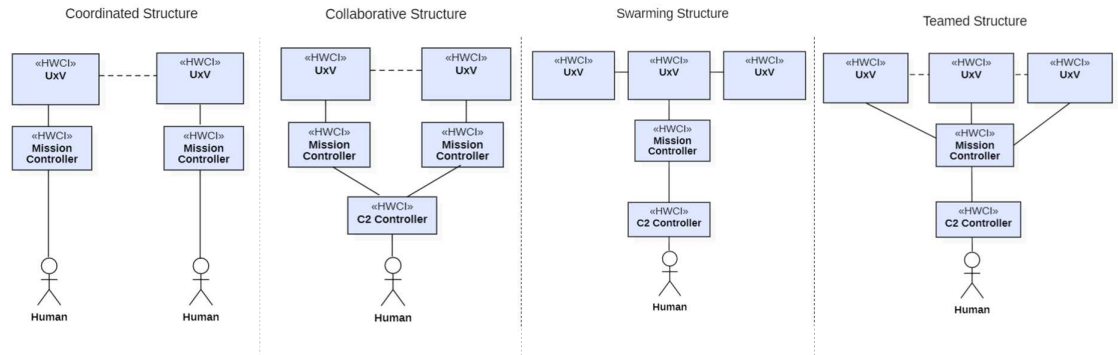
인터페이스를 담당하는 레이어로 구성되어 있다.

4.2.2 하드웨어 아키텍처

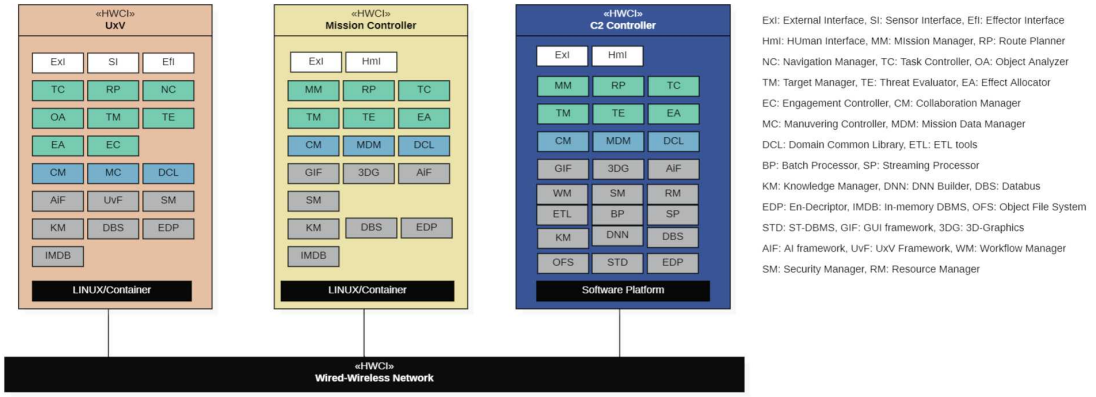
(그림 6)은 임무분산구조의 결정사항에 따라 4가지 형태의 하드웨어 아키텍처를 제시하고 있다. 실선의 연결은 특정 임무를 수행하기 위해 목적을 상호 공유하며 통제로 연결된 것을 의미하며, 점선의 연결은 통제가 아닌 단순 정보교환을 의미한다.

4.2.3 체계 아키텍처

(그림 7)은 하드웨어 형상항목에 소프트웨어 형상항목을 배치하여 하드웨어와 소프트웨어가 통합된 최종 체계 아키텍처를 보여주고 있다. 미래의 유무인 복합체계는 Zero-Trust Architecture가 보장된 유무선 네트워크를 통해 특정 임무를 수행하기 위해 상호 연동하며, 사이버 복원력을 갖고 MUM-T를 성공적으로 실현하는 자율화 소프트웨어 형상항목들로 임무를 완수하게 된다.



(그림 6) 참조 유무인 복합체계의 하드웨어 아키텍처



(그림 7) 참조 유무인 복합체계의 체계 아키텍처

5. 결 론

미래의 무기체계는 인공지능을 적용한 소프트웨어 중심의 유무인 복합체계로 발전하고 있다. Software-Defined Weapon을 넘어 Software-Defined Warfare로 발전할 것이며, 이러한 발전은 오히려 무기체계에 대한 사이버공격의 위협이 증가될 것으로 예상된다. 이러한 이유로 RMF라는 사이버 위협에 체계적으로 대응하기 위한 체계개발 요구가 증대되고 있다.

본 논문에서는 RMF와 같은 사이버 복원력을 요구하는 가이드라인이 부재한 상태에서도 품질을 확보할 수 있는 개발방법론을 적용하여 사이버 복원력을 고려한 유무인 복합체계의 참조 아키텍처를 제안하였다. 제안된 참조 아키텍처는 유무인 복합체계의 공통적인 기능의 식별과 신뢰성, 보안성, 유지보수 용이성 등 사이버 복원력에 영향을 줄 수 있는 품질속성을 기반으로 설계되었다. 따라서 본 논문에서 제안한 참조 아키텍처 및 개발방법은 사이버공격에 강건한 기술적 및 관리적 요소들을 포함하고 있어, 특정 유무인 복합체계 개발에 가이드라인을 제시할 수 있을 것으로 기대된다.

향후 연구로는 RMF를 기반으로 추가적인 보안통제항목을 기능적 및 비기능적 요구사항에 반영하여 개선된 아키텍처를 개발하는 것이다. 또한 유무인 복합체계는 인공지능을 적용하기 때문에 AI RMF[19]를 고려한 신뢰성(Trustworthiness)를 확보해야 하므로 이를 고려한 개선된 아키텍처의 개발도 향후에 연구해야 할 주제이다. 마지막으로 AI RMF를 고려할 때 제품의 품질속성 뿐만 아니라 데이터의 품질속성을 고려한 데이터 모델 및 아키텍처의 개발은 미래의 유무인 복합체계의 참조 아키텍처 연구에 또 다른 중요한 과제가 될 것으로 예상된다.

참 고 문 헌

[1] Rajesh Uppal, DARPA(AI, XAI, and AI next) Developing “Third Wave” AI based adaptive military systems that are trustworthy, learn continuously, and explain their rationale, International Defense, Security & Technology, May 2021.

[2] DARPA, DARPA Tiles Together a Vision of Mosaic Warfare, May 2018

[3] NIST, Risk Management Framework for Information Systems and Organizations: A

System Life Cycle Approach for Security and Privacy, NIST SP 800-37 Rev. 2, December 2018.
[4] Department of Defense, Military Standard Software Development and Documentation, MIL-STD-498, February 1994.
[5] ISO, Systems and software engineering — Software life cycle processes, ISO/IEC/IEEE 12207:2017, November 2017.
[6] 방위사업청, 무기체계 개발 및 관리 매뉴얼, 방위사업청 매뉴얼 제2020-1호, 2020년 2월.
[7] S. Lee, D. Kim, "Weapon Software: Challenge and Directions", ICSE 2020 Korean Co-located Events, July 2020.
[8] 김동환, 지능형 SW시스템 개발방법론, SDM-2023, 2023년 7월
[9] Object Management Group, OMG System Modeling Language Specification, OMG SysML 1.6, December 2019.
[10] Object Management Group, OMG Unified Modeling Language Specification, OMG UML 2.51, December 2017.
[11] Frans Osinga, Science, Strategy and War-The Strategy Theory of John Boyd, January 2005.
[12] ISO, Systems and software Quality Requirements and Evaluation, ISO/IEC/IEEE 25000:2014, March 2014.
[13] Michael Woudenberg, George Waltensperger, Troy Shideler, and Jerry Franke, "System Engineering of Autonomy: Frameworks for MUM-T Architecture", DSAIC Journals Vol. 7, No. 3, Summer 2020.
[14] Rob Woolley, Deploying Embedded Applications Faster with Containers, WNDRVR White paper, September 2021.
[15] Microsoft, The right secure hardware for your IoT Deployment, Microsoft White paper, November 2017.
[16] Rajive Joshi, Data-Oriented Architecture: A Loosely-Coupled Real-Time SOA, RTI White paper, August 2007.
[17] Vincenzo DiLuoffo, William R Michalson and Berk Sunar, "Robot Operating System 2: The need for a holistic security approach to robotic architectures", International Journal of Advanced Robotic Systems, Vol. 15, Issue 3, May 2018.
[18] Cloud Security Alliance, Software Defined Perimeter(SDP) and Zero Trust, CSA White Paper, 2020.
[19] NIST, Artificial Intelligence Risk Management Framework, NIST AI 100-1, January 2023.

저 자 약 력

김 동 환

이메일 : kimdonghwan@lignex1.com

- 1984년 인하대학교 전산학과 (학사)
- 1986년 KAIST 전산학과 (석사)
- 1998년 KAIST 정보및통신공학과 (박사과정수료)
- 1995년 전자계산기조직응용기술사
- 2008년 정보시스템수석감리원
- 1986년~1995년 국방과학연구소 선임연구원
- 1995년~1996년 (주)한조엔지니어링 부장
- 1996년~2000년 대우통신(주) 부장
- 2000년~2003년 팅크웨어(주) 상무
- 2003년~2009년 (주) 히어솔루션코리아 부사장
- 2010년~현재 LIG넥스원 C4ISTAR 연구개발2본부 연구위원
- 관심분야 : Dependable SW시스템 개발방법론 및 품질보증, 무인화 전투체계 및 자율주행 시스템